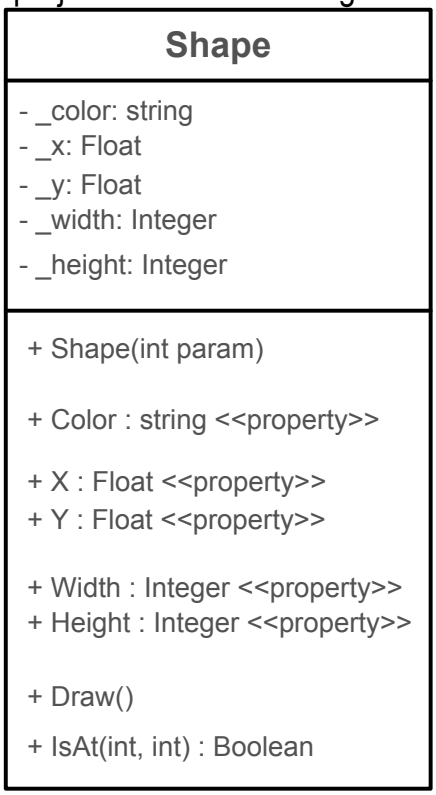
Task 2.2 Instructions

Over the course of the next few weeks you will develop a simple shape drawing program with Graphic User Interface. In this week, you begin by creating a *Shape* class. You should create this project separated from the previous task 2.1. In this way, we can easily extend the project in the following weeks.

1. Create a new ShapeDrawing project
2. Add a *Shape* class to your project. Use the following UML class diagram as a guide.



3. . The type *Color* is defined as a *string* at this stage. In Week 4, we will replace the datatype using SplashKit.

```
public class Shape
{
    private string _color; …
```

4. In the constructor, initialize ***_color*** to a string *"Color. Azure"* if the first letter of your first name is from A to L. Otherwise, set by *"Color.Chocolate"*. In that constructor, set ***_x*** and ***_y*** to 0.0f (the suffix f makes . a float value), and both the _width and _height are assigned to the value of a given parameter called param. When you later create objects based on this Shape class, please specify the parameter to be 1 , where is the last two digits of your student ID.

5. The ***Draw*** method will print out the essential information including the shape's color, position, and dimension. In Week 4, we will replace this method with a drawing function from SplashKit

```
public class Shape
{
    …
    public void Draw()
    {
       Console.WriteLine("Color is " + _color);
       Console.WriteLine("Position X is " + _x);
          ....
    }
```

6. Add the ***IsAt*** method which takes ***two*** integers (int xInput, int yInput) representing a point in 2d space - like a point on the screen), and returns a Boolean to indicate if the shape is at that point. You need to return true if the point *pt* is within the shape's area (as defined by the shape's coordinates).

Tip: What does it mean for a point to be considered inside the area of a rectangle?Assuming, (x1,y1) and (x2,y2) are the top-left and bottom-right corners of a rectangle, respectively. A 2Dpoint (xInput, yInput) is inside the rectangle only if (xInput > x1 && xInput < x2 && yInput > y1 && yInput < y2)

7. Add all properties (as defined in the UML diagram) to class *Shape*.
8. Return to the *Program.cs* file.
9. In ***Main***,
    - Add a ***myShape*** local variable of the type *Shape*.
    - Assign ***myShape***, a **new** *Shape* object using the Shape constructor
    - Tell myShape to Draw itself
    - Compile and run to obtain the result from the console
10. Save and backup your code using Cloud, USB, or any drive

When you arrive at your lab, you will receive the verification tasks.
See you very soon. I